Engagement Stage: Cross-Selling and Upselling

After identifying and targeting potential new customers through look-alike or uplift modeling, the next step is to enhance engagement by focusing on cross-selling and upselling opportunities. This stage aims to deepen the relationship with existing customers by encouraging them to purchase additional products or services.

Cross-Selling and Upselling

Objective:

- Increase the average revenue per customer by recommending relevant additional products or services based on customer needs and behavior.

Methodology:

1. **Data Preparation:**
    - Collect and preprocess customer data to ensure it is clean and suitable for analysis.
    - Data sources include transaction history, product holdings, demographic information, and engagement metrics.
2. **Attribute Selection:**
    - **Raw Attributes:**
        - Age
        - Income
        - Occupation
        - Credit score (CIBIL score)
        - Existing product holdings (e.g., types of accounts, loans, credit cards)
        - Transaction history (frequency, recency, monetary value)
        - Customer interaction data (e.g., customer service interactions, website visits)

- **Derived Attributes:**
    - Product affinity score: Likelihood of interest in a specific product based on past behavior and similar customer profiles.
    - Engagement score: Measure of overall engagement with the bank's services.
    - Transaction trends: Patterns in spending and saving behavior.
    - Churn propensity: Likelihood of customer attrition.
    - Lifetime value (LTV): Projected long-term value of the customer to the bank.
- **Target Variables:**
    - Response to cross-sell or upsell offers (binary: yes/no)
    - Purchase of additional products (e.g., loans, credit cards)

3. **Model Selection:**
    - **Recommendation System:** Collaborative Filtering, Content-Based Filtering, or Hybrid models.
    - **Machine Learning Algorithms:** Random Forest, Gradient Boosting Machines (GBM), Neural Networks.

Implementation Steps

1. **Data Collection and Preparation:**

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder, Standar
dScaler

# Sample customer data
data = {
    'age': [25, 45, 35, 50, 23],
    'income': [50000, 120000, 75000, 100000, 45000],
    'cibil_score': [700, 800, 750, 780, 690],
    'existing_products': ['savings', 'loan', 'credit car
d', 'savings', 'loan'],
    'transaction_frequency': [15, 50, 25, 30, 10],
```

```
    'average_transaction_value': [2000, 5000, 3000, 400
0, 1500],
    'engagement_score': [3, 5, 4, 4, 2],
    'response_to_cross_sell': [1, 0, 1, 0, 1]
}
df = pd.DataFrame(data)

# Encode categorical variables
encoder = OneHotEncoder()
encoded_products = encoder.fit_transform(df[['existing_p
roducts']]).toarray()
df = df.join(pd.DataFrame(encoded_products, columns=enco
der.get_feature_names_out(['existing_products']))).drop
('existing_products', axis=1)

# Standardize numerical features
scaler = StandardScaler()
numerical_features = ['age', 'income', 'cibil_score', 't
ransaction_frequency', 'average_transaction_value', 'eng
agement_score']
df[numerical_features] = scaler.fit_transform(df[numeric
al_features])
```

2. **Model Training and Evaluation:**

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report

# Split data into features and target variable
X = df.drop(['response_to_cross_sell'], axis=1)
y = df['response_to_cross_sell']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X,
y, test_size=0.2, random_state=42)

# Train Random Forest model
model = RandomForestClassifier(n_estimators=100, random_
```

```
state=42)
model.fit(X_train, y_train)

# Predict and evaluate
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))
```

3. **Recommendation System (Alternative Approach):**

   Collaborative Filtering Example:

```
from sklearn.neighbors import NearestNeighbors
import numpy as np

# Sample matrix of customer-product interactions
interaction_matrix = np.array([
    [1, 1, 0, 0, 0],
    [1, 0, 1, 0, 0],
    [0, 1, 1, 0, 0],
    [0, 0, 0, 1, 1],
    [1, 0, 0, 1, 0]
])

# Train a KNN model for collaborative filtering
model_cf = NearestNeighbors(n_neighbors=2, algorithm='au
to')
model_cf.fit(interaction_matrix)

# Find similar customers
customer_index = 0  # Example customer index
distances, indices = model_cf.kneighbors(interaction_mat
rix[customer_index].reshape(1, -1), n_neighbors=2)
similar_customers = indices.flatten()

print(f"Customers similar to customer {customer_index}:
{similar_customers}")
```

4. **Deploy and Monitor:**

- Deploy the trained models to the bank's marketing system.
- Monitor customer responses to cross-sell and upsell offers.
- Continuously refine the models based on new data and feedback.

Variables for Cross-Selling and Upselling

1. **Customer Attributes:**
    - **Age:** Influences financial needs and product preferences.
    - **Income:** Indicates financial capacity and potential product interest.
    - **Occupation:** Provides insights into lifestyle and financial requirements.
    - **Credit Score:** Reflects creditworthiness and financial behavior.
    - **Existing Products:** Shows current product holdings and potential gaps.
2. **Transactional and Behavioral Data:**
    - **Transaction Frequency:** Indicates engagement level with banking services.
    - **Average Transaction Value:** Suggests spending capacity and behavior.
    - **Engagement Score:** Measures overall interaction and activity with the bank.
    - **Response to Previous Campaigns:** Historical data on response rates to past marketing efforts.
3. **Derived Attributes:**
    - **Product Affinity Score:** Calculated based on similarity to other customers who have purchased the product.
    - **Churn Propensity:** Probability of the customer leaving the bank.
    - **Lifetime Value (LTV):** Estimated total value the customer will bring to the bank over time.
4. **Target Variables:**
    - **Response to Cross-Sell/Upsell Offers:** Binary indicator of whether the customer accepted the offer.
    - **Additional Product Purchases:** Details of additional products purchased as a result of cross-selling or upselling efforts.

Conclusion

By implementing a robust cross-selling and upselling strategy using advanced machine learning techniques, the bank can significantly enhance customer engagement, increase revenue per customer, and build long-term customer relationships. This approach ensures personalized and effective marketing efforts, aligning with customer needs and preferences, ultimately driving growth and customer satisfaction.